# Tensor-variate Restricted Boltzmann Machines

**Tu Dinh Nguyen†, Truyen Tran†‡, Dinh Phung†, Svetha Venkatesh†**
†Center for Pattern Recognition and Data Analytics
School of Information Technology, Deakin University, Geelong, Australia
‡Institute for Multi-Sensor Processing and Content Analysis
Curtin University, Australia
{ngtu,truyen.tran,dinh.phung,svetha.venkatesh}@deakin.edu.au

## Abstract

Restricted Boltzmann Machines (RBMs) are an important class of latent variable models for representing vector data. An under-explored area is multimode data, where each data point is a matrix or a tensor. Standard RBMs applying to such data would require vectorizing matrices and tensors, thus resulting in unnecessarily high dimensionality and at the same time, destroying the inherent higher-order interaction structures. This paper introduces Tensor-variate Restricted Boltzmann Machines (TvRBMs) which generalize RBMs to capture the multiplicative interaction between data modes and the latent variables. TvRBMs are highly compact in that the number of free parameters grows only linear with the number of modes. We demonstrate the capacity of TvRBMs on three real-world applications: handwritten digit classification, face recognition and EEG-based alcoholic diagnosis. The learnt features of the model are more discriminative than the rivals, resulting in better classification performance.

## 1 Introduction

Restricted Boltzmann Machines (RBMs) are important generative models that are capable of modeling diverse data types and affording fast inference and efficient MCMC-based learning procedures (Smolensky 1986; Hinton 2002; Salakhutdinov and Hinton 2009c; Tieleman and Hinton 2009; Ranzato, Krizhevsky, and Hinton 2010; Memisevic and Hinton 2010). RBMs, crucially, can serve as building blocks for deep architectures (Hinton and Salakhutdinov 2006; Salakhutdinov and Hinton 2009a).

RBMs, however, have been largely designed for vector data. Consider EEG recordings wherein each trial collects signals from multiple channels. Each channel data can be represented as a 2D $\langle \text{time}, \text{frequency} \rangle$ spectrogram. The $\langle \text{channel}, \text{time}, \text{frequency} \rangle$ data can be represented as a 3-mode tensor. Applying RBMs on this data has two problems: first, this data needs to be flattened into a linear vector (see Fig. 1) - thereby, the vectorization breaks the explicit multimode structures, losing vital information about interactions across modes. Second, this flattened data representation leads to a increase in number of parameters to be estimated, requiring solutions that are not optimal. For a typically EEG trial, the number of spectrogram pixels could be excessively large even for short observation periods and low-sampling rates - as example, a $64 \times 64 \times 64$ $\langle \text{channel}, \text{time}, \text{frequency} \rangle$ tensor has $262,144$ pixels). Each patient has multiple episodes, and each episode is a tensor. What is required is a modeling of distribution over $\text{N}$-mode tensor space, for $\text{N} \geq 2$. Our intuition to do this is to model the associative weights between the structured data and the hidden units as a $(\text{N}+1)$-mode tensor. The tensor decomposition is learnt automatically. No previous work has explored the use of RBM for tensor data.

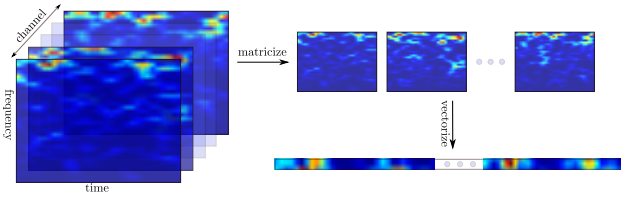


Figure 1: An EEG trial represented as a tensor of $\langle \text{channel}, \text{time}, \text{frequency} \rangle$ and its the matricization and vectorization.

We propose *Tensor-variate Restricted Boltzmann Machine* (TvRBM) that generalizes the RBM to capture interactions among data modes. We employ multiway factoring of mapping parameters that link data modes and hidden units. Modes interact in a multiplicative fashion gated by hidden units. Using a tensor data and the hidden layer to construct a $(\text{N}+1)$-mode tensor, we start from a similar decomposition to (Memisevic and Hinton 2010; Taylor and Hinton 2009; Ranzato, Krizhevsky, and Hinton 2010), but tie the parameters across each mode of variation. Efficient parameter estimation is derived for this model. This model has the following advantages: first, the number of mapping parameters grows linearly with the number of modes rather than exponentially. Second, the intra-layer conditional independence is retained and thus not affect the efficiency of model parameter learning. We demonstrate the capacity of our proposed model through comprehensive experiments on three diverse real-world datasets: handwritten digits, facial images, and EEG signals. The experimental results show that the classification performance of the TvRBM is more competitive than the standard RBM and existing tensor decomposition methods. Our main contributions are: (i) a new way of com-

puting distribution over tensor space using RBM and (ii) a comprehensive evaluation of the effectiveness of our method on three applications: handwritten digit classification, face recognition and alcoholic diagnosis using EEG signals.

The rest of paper is organized as follows. Sec. 2 presents an insightful review of the related literature. We then describe the standard RBM and tensor notations in Sec. 3, followed by the model presentation and parameter learning of our Tensor-variate RBM in Sec. 4. The experimental results on three applications are then reported in Sec. 5. Finally, Sec. 6 concludes the paper.

## 2 Related Work

The multiway data modeling problem is well-studied. The most popular approach is to treat the entire dataset as a $(\mathrm{N}+1)$-mode tensor and apply tensor decomposition methods. Two most well-known techniques are the Tucker decomposition (Tucker 1963) and PARAFAC (Carroll and Chang 1970). More recently, variants have been introduced, including 2D nonnegative matrix factorization (2DNMF) (Zhang, Chen, and hua Zhou 2005), nonnegative Tucker decomposition (Kim and Choi 2007), and nonnegative tensor factorization (Shashua and Hazan 2005). These are linear and do not generalize to unseen data because data distribution is not modeled. When seeing a new data point, these methods must perform an expensive "fold-in" procedure to estimate projection. Probabilistic interpretation has subsequently been introduced, notably the probabilistic Tucker decomposition (Chu and Ghahramani 2009) and its nonparametric Bayesian extension (Xu, Yan, and Qi 2012). These models are directed whilst the TvRBM is undirected with log-linear parameterization.

For data points in the matrix form (e.g., 2D images), well-known methods include 2D principal components analysis (2DPCA) (Yang et al. 2004) and 2D linear discriminant analysis (2DLDA) (Ye, Janardan, and Li 2004). These are also linear methods with strong assumptions of Gaussian noise. By contrast, our TvRBM is a probabilistic model that learns a probabilistic non-linear mapping from data to the representation space. More closely related to the generative nature of the RBM is factor analysis (FA), which is a directed model of vector data. Matrix and tensor extensions have been introduced in recent years: matrix-variate FA (MVFA) (Xie et al. 2008), and the tensor analyzer (Tang, Salakhutdinov, and Hinton 2013).

Multiplicative interactions and multiway factoring have been studied within the RBM community to capture the pairwise association between vectors and the gating effect (Memisevic and Hinton 2010; Taylor and Hinton 2009; Ranzato, Krizhevsky, and Hinton 2010), as well as factors of variation (Reed and Lee 2013). Although the mathematics of decomposing the $(\mathrm{N}+1)$-order mapping tensor $\mathscr{W}$ between their work and ours has similar tensor factorization form, the nature of data and the purposes are entirely different. They examine the Cartesian product of two vectors, but the input data is still vector, not tensor and there does not exist the concept of "data mode" (e.g., channel, time, frequency). More specially, the pairwise interaction between pixel elements is considered in (Ranzato, Krizhevsky, and Hinton 2010), but the pixels are represented as a vector. In (Memisevic and Hinton 2007; 2010), the transformation between two image vectors is studied, but two similar vectors do not constitute two orthogonal modes such as time and frequency. Second, generalizing pairwise transformation to three or more vectors proves to be difficult. On the other hand, we study the data in a high-order tensor of orthogonal modes. The $(\mathrm{N}+1)$-order tensor factorization in our case reveals the structure of the interaction among N modes, while it is the higher-order interaction between vector elements in previous work.

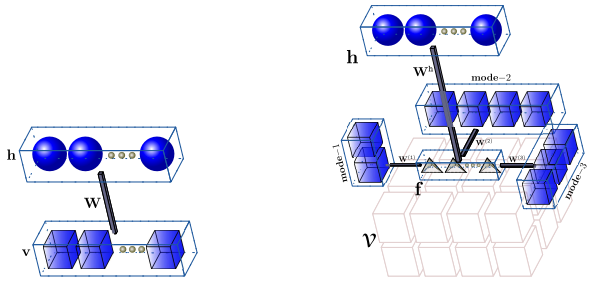


Figure 2: Graphical illustrations of RBM (left) and TvRBM (right). The cubic nodes are observed, the sphere nodes are latent. The TvRBM models the 3D input data of $2 \times 4 \times 3$ and the triangular pyramids represent 4-way factors.

## 3 Preliminaries

### 3.1 Representing vector data using RBM

Denote by $\mathbf{v} = (\mathrm{v}_1, \mathrm{v}_2, ..., \mathrm{v}_\mathrm{M})$ the variable representing the visible data. Our goal is to learn a higher, latent representation $\mathbf{h} = (\mathrm{h}_1, \mathrm{h}_2, ..., \mathrm{h}_\mathrm{K}) \in \{0,1\}^\mathrm{K}$. A Restricted Boltzmann Machine (RBM) (Smolensky 1986; Hinton and Salakhutdinov 2006) is a bipartite undirected graphical model encoding these two layers (see Fig. 2 (left) for a graphical illustration). The RBM assigns energy for a joint configuration $(\mathbf{v}, \mathbf{h})$ as:

$$E(\mathbf{v}, \mathbf{h}; \psi) = -\left[\mathcal{F}(\mathbf{v}) + \mathbf{a}^\top \mathbf{v} + \mathbf{b}^\top \mathbf{h} + \mathbf{v}^\top \mathbf{W} \mathbf{h}\right] \quad (1)$$

where $\mathbf{a} \in \mathbb{R}^\mathrm{M}$, $\mathbf{b} \in \mathbb{R}^\mathrm{K}$ are the biases, $\mathbf{W} \in \mathbb{R}^{\mathrm{M} \times \mathrm{K}}$ is the mapping parameters, and $\mathcal{F}(\mathbf{v})$ is type-specific function (e.g., $\mathcal{F}(\mathbf{v}) = 0$ for binary input and $\mathcal{F}(\mathbf{v}) = -0.5 \sum_m \mathrm{v}_m^2$ for Gaussian variables). The model admits the Boltzmann distribution: $p(\mathbf{v}, \mathbf{h}; \psi) \propto \exp[-E(\mathbf{v}, \mathbf{h}; \psi)]$, where $\psi = \{\mathbf{a}, \mathbf{b}, \mathbf{W}\}$ is model parameter.

The bipartite structure of RBM enables units in one layer become conditionally independent given the other layer. Thus the conditional distributions over hidden and visible units are factorized as:

$$p(\mathbf{v} \mid \mathbf{h}; \psi) = \prod_{m=1}^{\mathrm{M}} p(\mathrm{v}_m \mid \mathbf{h})$$

$$p(\mathbf{h} \mid \mathbf{v}; \psi) = \prod_{k=1}^{\mathrm{K}} p(\mathrm{h}_k \mid \mathbf{v})$$

These provide fast layer-wise MCMC sampling, leading to efficient stochastic gradient learning, such as Contrastive Divergence (Hinton 2002) and its variant (Tieleman and Hinton 2009).

### 3.2 Tensor notations

Following (Kolda and Bader 2009), we use plain lowercase letters (e.g., t) to indicate scalar values; bold lowercase letters (e.g., $\mathbf{t}$) for vectors; bold uppercase letters for matrices (e.g., $\mathbf{T}$); Euler script letters for higher-order tensor, e.g. the N-mode tensor: $\mathscr{T} \in \mathbb{R}^{\mathrm{D}_{1:\mathrm{N}}}$ where $\mathrm{D}_{1:\mathrm{N}} \triangleq \mathrm{D}_1 \times \mathrm{D}_2 \times ... \times \mathrm{D}_\mathrm{N}$ is the product space over N dimensions. Denote by $\mathbf{t}_{\bullet i}$ the $i$-th column vector of matrix $\mathbf{T}$. The symbol "$\circ$" denotes the vector outer product, e.g., the rank-one N-mode tensor can be written as the outer product of N vectors:

$$\mathscr{T} = \mathbf{x}^{(1)} \circ \mathbf{x}^{(2)} \circ ... \circ \mathbf{x}^{(\mathrm{N})}$$

Let $\bar{\times}_n$ indicate the $n$-th mode product which is the multiplication of a vector with a tensor along mode $n$, resulting in a $(\mathrm{N}-1)$-mode tensor:

$$\mathscr{X} = \mathscr{T} \bar{\times}_n \mathbf{t}$$

where $\mathbf{t} \in \mathbb{R}^{\mathrm{D}_n}$, $\mathscr{X} \in \mathbb{R}^{\mathrm{D}_{\neg n}}$ and $\mathrm{D}_{\neg n} \triangleq \mathrm{D}_{\{1:\mathrm{N}\}\backslash n}$ denotes the product space over N dimensions excluding $\mathrm{D}_n$.
The inner product of two same-sized tensors $\mathscr{X}, \mathscr{Y} \in \mathbb{R}^{\mathrm{D}_{1:\mathrm{N}}}$ is the sum of their element-wised products:

$$\langle \mathscr{X}, \mathscr{Y} \rangle = \sum_{d_1=1}^{\mathrm{D}_1} \sum_{d_2=1}^{\mathrm{D}_2} ... \sum_{d_\mathrm{N}=1}^{\mathrm{D}_\mathrm{N}} \mathrm{x}_{d_1 d_2 ... d_\mathrm{N}} \mathrm{y}_{d_1 d_2 ... d_\mathrm{N}}$$

## 4 Tensor-variate Restricted Boltzmann Machines

We now describe the Tensor-variate RBM (TvRBM) for multimode data. See Fig. 2 (right) for an example of TvRBM to model 3D data.

### 4.1 Model definition

In TvRBM, the set of visible units is represented by a N-mode tensor $\mathscr{V} \in \mathbb{R}^{\mathrm{D}_{1:\mathrm{N}}}$ and the hidden units are the same as in RBM. The goal is to model the joint distribution $p(\mathscr{V}, \mathbf{h})$. The energy in Eq. (1) turns into the following form:

$$E(\mathscr{V},\mathbf{h}) = -\left[\mathcal{F}(\mathscr{V}) + \langle \mathscr{A}, \mathscr{V} \rangle + \mathbf{b}^\top \mathbf{h} + \langle \mathscr{V}, \mathscr{W} \bar{\times}_{\mathrm{N}+1} \mathbf{h} \rangle\right] \quad (2)$$

where $\mathcal{F}(\mathscr{V})$ is type-specific function, $\mathscr{A} \in \mathbb{R}^{\mathrm{D}_{1:\mathrm{N}}}$ are visible biases, and $\mathscr{W} \in \mathbb{R}^{\mathrm{D}_{1:\mathrm{N}} \times \mathrm{K}}$ are mapping parameters. The hidden posterior is

$$p(\mathrm{h}_k = 1 \mid \mathscr{V}) = \sigma\left[\mathrm{b}_k + \left\langle \mathscr{V}, \mathscr{W} \bar{\times}_{\mathrm{N}+1} \mathbf{1}_k^\mathrm{K} \right\rangle\right] \quad (3)$$

where $\mathbf{1}_k^\mathrm{K}$ is one-hot representation of K-length vector with all zeros but 1 at $k^{\text{th}}$ position, $\sigma(\mathrm{x}) = (1+e^{-x})^{-1}$ is the sigmoid function. The generative distribution $p(\mathscr{V} \mid \mathbf{h})$, on the other hand, is type-specific. In what follows, we present the most popular cases, namely the binary and Gaussian inputs, but the generalization to Poisson (Salakhutdinov and Hinton 2009b), multinomial (Salakhutdinov and Hinton 2009c), and mixed types (Tran, Phung, and Venkatesh 2013) is omitted. Let:

$$\mathcal{G}_{d_1 d_2 ... d_\mathrm{N}}(\mathbf{h}) = \left[\mathscr{W} \bar{\times}_1 \mathbf{1}_{d_1}^{\mathrm{D}_1} \bar{\times}_2 \mathbf{1}_{d_2}^{\mathrm{D}_2} \bar{\times}_3 ... \bar{\times}_\mathrm{N} \mathbf{1}_{d_\mathrm{N}}^{\mathrm{D}_\mathrm{N}}\right]^\top \mathbf{h}$$

- For binary input, we have $\mathcal{F}(\mathscr{V}) = 0$ (in Eq. (2)) and the following generative distribution:

$$p(\mathrm{v}_{d_1 d_2 ... d_\mathrm{N}} = 1 \mid \mathbf{h}) = \sigma\left[\mathrm{a}_{d_1 d_2 ... d_\mathrm{N}} + \mathcal{G}_{d_1 d_2 ... d_\mathrm{N}}(\mathbf{h})\right]$$

- For Gaussian input, assuming unit variance, i.e., $\mathcal{F}(\mathscr{V}) = -0.5\langle \mathscr{V}, \mathscr{V} \rangle$, the generative distribution reads

$$p(\mathrm{v}_{d_1 d_2 ... d_\mathrm{N}} | \mathbf{h}) = \mathcal{N}\left[\{\mathrm{a}_{d_1 d_2 ... d_\mathrm{N}} + \mathcal{G}_{d_1 d_2 ... d_\mathrm{N}}(\mathbf{h})\}; \mathbf{1}^{\mathrm{D}_{1:\mathrm{N}}}\right]$$

  in which $\mathbf{1}^{\mathrm{D}_{1:\mathrm{N}}}$ is the tensor where all elements are 1.

Once all parameters are fully specified, the new representation of an input data can be obtained by projecting onto hidden space $\hat{\mathbf{h}} = \left(\hat{\mathrm{h}}_1, \hat{\mathrm{h}}_2, ..., \hat{\mathrm{h}}_K\right)$, where $\hat{\mathrm{h}}_k = p(\mathrm{h}_k = 1 \mid \mathscr{V})$ as in Eq. (3). The higher representation can be used as input for further classification tasks.

### 4.2 (N+1)-way factoring of multiplicative interactions

A major problem with the parameterization in Eq. (2) is the excessively large number of free parameters which scales as the product of data mode and hidden dimensions. In particular, the $(\mathrm{N}+1)$-mode mapping tensor $\mathscr{W}$ has $\mathrm{K}\prod_\mathrm{n} \mathrm{D}_n$ elements, which quickly reaches billions when the mode dimensionalities K, $\mathrm{D}_{1:\mathrm{N}}$ and N are moderate. This makes learning extremely difficult for several reasons. First, it would require a large dataset for a robust estimate of parameters. Second, it is hard to control the bounding of hidden activation values $\left\langle \mathscr{V}, \mathscr{W} \bar{\times}_{\mathrm{N}+1} \mathbf{1}_k^\mathrm{K} \right\rangle$ in Eq. (3). Thus the hidden posteriors are easily collapsed into either 0 or 1 and no more learning occurs. Finally, the model requires $\mathcal{O}\left(\mathrm{K}\prod_\mathrm{n} \mathrm{D}_\mathrm{n}\right)$ memory for parameters.

To this end, we employ $(\mathrm{N}+1)$-way factoring (Memisevic and Hinton 2010) to construct the multiplicative interactions between visible modes and hidden units. With F factors, the parameter tensor $\mathscr{W}$ is decomposed using the Kruskal operator as follows:

$$\begin{aligned}\mathscr{W} &= [\![\boldsymbol{\lambda}; \mathbf{W}^{(1)}, ..., \mathbf{W}^{(\mathtt{N})}, \mathbf{W}^{\mathtt{h}}]\!] \\ &= \sum_{\mathrm{f}=1}^{\mathrm{F}} \lambda_\mathrm{f} \mathbf{w}_{\bullet \mathrm{f}}^{(1)} \circ ... \circ \mathbf{w}_{\bullet \mathrm{f}}^{(\mathtt{N})} \circ \mathbf{w}_{\bullet \mathrm{f}}^{\mathtt{h}}\end{aligned}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^\mathrm{F}$ is the scaling vector, the matrix $\mathbf{W}^{(\mathtt{n})} \in \mathbb{R}^{\mathrm{D}_n \times \mathrm{F}}$ represents the mode-factor weights, and $\mathbf{W}^{\mathtt{h}} \in \mathbb{R}^{\mathrm{K} \times \mathrm{F}}$ the hidden-factor. Here we fix $\boldsymbol{\lambda} = \mathbf{1}$ for simplicity, so we obtain:

$$\mathrm{w}_{d_1 d_2 ... d_\mathrm{N} k} = \sum_{\mathrm{f}=1}^{\mathrm{F}} \sum_{d_1 d_2 ... d_\mathrm{N} k} \mathrm{w}_{d_1 \mathrm{f}}^{(1)} ... \mathrm{w}_{d_\mathrm{N} \mathrm{f}}^{(\mathtt{N})} \mathrm{w}_{k\mathrm{f}}^{\mathtt{h}}$$

This factoring allows multiplicative interactions between modes to be moderated by the hidden units through the hidden-factor matrix $\mathbf{W}^{\mathtt{h}}$. Thus the model captures the

modes of variation through the new representation by $\mathbf{h}$. The number of mapping parameters is drastically reduced to $\mathrm{F}\left(\mathrm{K}+\sum_n \mathrm{D}_n\right)$, which grows linearly rather than exponentially in N. The memory space requirement decreases accordingly. Importantly, the presence of the decomposition does not lead to chicken-and-egg problem that can make learning and inference difficult. More specifically, the conditional independence among intra-layer variables, i.e., $p\left(\mathbf{h} \mid \mathscr{V}\right) = \prod_{k=1}^{\mathrm{K}} p\left(\mathrm{h}_k \mid \mathscr{V}\right)$ and $p\left(\mathscr{V} \mid \mathbf{h}\right) = \prod_n \prod_{d_n=1}^{\mathrm{D}_n} p\left(\mathrm{v}_{d_1 d_2 \ldots d_{\mathrm{N}}} \mid \mathbf{h}\right)$, are not affected. Therefore the proposed model preserves fast sampling and inference properties of RBM.

### 4.3 Parameter learning

Denote by $\phi = \left\{\mathscr{A}, \mathbf{b}, \mathbf{W}^{(\mathtt{1})}, \mathbf{W}^{(\mathtt{2})} ..., \mathbf{W}^{(\mathtt{N})}, \mathbf{W}^{\mathtt{h}}\right\}$ the model parameters. Learning seeks the maximizer of the data likelihood: $\mathcal{L}\left(\mathscr{V}; \phi\right) = p\left(\mathscr{V}; \phi\right) = \sum_{\mathbf{h}} p\left(\mathscr{V}, \mathbf{h}; \phi\right)$ with respect to the parameter $\phi$. The gradient of the log-likelihood is the difference of two expectations:

$$\frac{\partial}{\partial \phi} \log \mathcal{L}\left(\mathscr{V}; \phi\right) = \mathbb{E}_{\mathscr{V}, \mathbf{h}}\left[\frac{\partial E\left(\mathscr{V}, \mathbf{h}\right)}{\partial \phi}\right] - \mathbb{E}_{\mathbf{h} \mid \mathscr{V}}\left[\frac{\partial E\left(\mathscr{V}, \mathbf{h}\right)}{\partial \phi}\right]$$

where $\mathbb{E}_{\mathscr{V}, \mathbf{h}}$ is the model expectation w.r.t. $p\left(\mathscr{V}, \mathbf{h}\right)$ and $\mathbb{E}_{\mathbf{h} \mid \mathscr{V}}$ is the data expectation w.r.t. $p\left(\mathbf{h} \mid \mathscr{V}\right)$. The data expectation can be computed efficiently using Eq. (3) whilst the model expectation is intractable due to the sum over exponential space of hidden units. Fortunately, the Contrastive Divergence procedure (Hinton 2002) allows fast approximation using short Markov chains. Starting from observed data, the chains collect samples by alternating between $\hat{\mathscr{V}} \sim P\left(\mathscr{V} \mid \hat{\mathbf{h}}\right)$ and $\hat{\mathbf{h}} \sim P\left(\mathbf{h} \mid \hat{\mathscr{V}}\right)$.

The derivatives with respect to mode- and hidden-factor matrices are:

$$\frac{\partial}{\partial \mathrm{w}^{(\mathtt{n})}_{d_n \mathrm{f}}} E\left(\mathscr{V}, \mathbf{h}; \phi\right) = -\left(\mathbf{h}^\top \mathbf{w}^{\mathtt{h}}_{\bullet \mathrm{f}}\right) \left\langle \mathscr{X}, \mathscr{Y} \right\rangle$$

$$\frac{\partial}{\partial \mathrm{w}^{\mathtt{h}}_{k \mathrm{f}}} E\left(\mathscr{V}, \mathbf{h}; \phi\right) = -\mathrm{h}_k \left\langle \mathscr{V}, \mathbf{w}^{(\mathtt{1})}_{\bullet \mathrm{f}} \circ ... \circ \mathbf{w}^{(\mathtt{N})}_{\bullet \mathrm{f}} \right\rangle$$

where $\mathscr{X} = \mathscr{V} \bar{\times}_{\mathrm{n}} \mathbf{1}^{\mathrm{D}_n}_{d_n}$ and $\mathscr{Y} = \mathbf{w}^{(\mathtt{1})}_{\bullet \mathrm{f}} \circ ... \circ \mathbf{w}^{(\mathtt{n}-\mathtt{1})}_{\bullet \mathrm{f}} \circ \mathbf{w}^{(\mathtt{n}+\mathtt{1})}_{\bullet \mathrm{f}} \circ ... \circ \mathbf{w}^{(\mathtt{N})}_{\bullet \mathrm{f}}$.

Finally the parameters can be updated using stochastic gradient ascent as: $\phi \leftarrow \phi + \eta \left(\frac{\partial}{\partial \phi} \log \mathcal{L}\left(\mathscr{V}; \phi\right)\right)$ for a learning rate $\eta > 0$. However, for numerical stability, we employ an adaptive rate for factor matrices so that the update step size is bounded (see Sec. 5.1 for implementation details).

### 4.4 Receptive fields visualization

For the data which have spatial structure such as images, one of the attractive capabilities of RBMs is that the hidden units can discover interpretable features which are visualized by displaying their learnt receptive fields. The receptive field of each hidden unit is formed by the weights connecting that hidden unit to the visible units. These also enable the visual assessment of the model generalization quality. However, there are no explicit connection weights between hidden units and visible ones in the TvRBM. We estimate these weights from the multiplicative interactions. Assuming that the spatial features lie at $n$-th mode of the data, the learnt filters are matrix $\mathbf{R}$ given by:

$$\mathbf{R} = \sum_{\mathrm{f}=1}^{\mathrm{F}} \mathrm{x}_{\mathrm{f}} \mathrm{y}_{\mathrm{f}} \left(\mathbf{w}^{\mathtt{h}}_{\bullet \mathrm{f}} \mathbf{w}^{(\mathtt{n})\top}_{\bullet \mathrm{f}}\right) \tag{4}$$

$$\text{with } \mathrm{x}_{\mathrm{f}} = \left(\mathbf{1}^{\mathrm{D}_1^\top} \mathbf{w}^{(\mathtt{1})}_{\bullet \mathrm{f}}\right) ... \left(\mathbf{1}^{\mathrm{D}_{n-1}^\top} \mathbf{w}^{(\mathtt{n}-\mathtt{1})}_{\bullet \mathrm{f}}\right)$$

$$\text{and } \mathrm{y}_{\mathrm{f}} = \left(\mathbf{1}^{\mathrm{D}_{n+1}^\top} \mathbf{w}^{(\mathtt{n}+\mathtt{1})}_{\bullet \mathrm{f}}\right) ... \left(\mathbf{1}^{\mathrm{D}_{\mathrm{N}}^\top} \mathbf{w}^{(\mathrm{N})}_{\bullet \mathrm{f}}\right)$$

## 5 Implementation and Results

In this section, we evaluate the Tensor-variate RBM on three real-world applications of very different natures: handwritten digit classification, face recognition and alcoholic diagnosis from EEG signals. Our main goal is to demonstrate the power of TvRBM in learning robust representations for high-dimensional data with several modes of variation and limited number of training samples.

### 5.1 Implementation

We use binary visible units for image data and Gaussian ones for EEG signals. Following (Hinton and Salakhutdinov 2006), the pixel intensities of images are normalized into the range $[0, 1]$ and treated as empirical probabilities of the input.

As factors interact in a multiplicative manner leading to a potential numerical instability in gradient, a sensible parameter initialization and control of step size in parameter update would stabilize the learning and achieve faster convergence. The factorized parameters are initialized randomly from $\mathcal{N}\left(0; 0.1\right)$. We initialize the visible biases using the first moment matching:

$$\mathscr{A} = \begin{cases} \log\left(\bar{\mathscr{V}}\right) - \log\left(1 - \bar{\mathscr{V}}\right) - \mathscr{W} \bar{\times}_{\mathrm{N}+1} \mathbf{h}_0 & \\ & \text{if } \mathrm{v} \in \{0, 1\} \\ \bar{\mathscr{V}} - \mathscr{W} \bar{\times}_{\mathrm{N}+1} \mathbf{h}_0 & \text{if } \mathrm{v} \in \mathbb{R} \end{cases}$$

where $\mathbf{h}_0$ is drawn randomly, and $\bar{\mathscr{V}}$ is the mean of the input over the dataset. The hidden bias is initialized as $\mathrm{b}_k = -\left\langle \bar{\mathscr{V}}, \mathscr{W} \bar{\times}_{\mathrm{N}+1} \mathbf{1}^{\mathrm{K}}_k \right\rangle$. For binary models, the step size of is fixed to 0.01. Hidden, visible and mode-factor learning rates are fixed to 0.1. In the Gaussian models, the step size and learning rates are 10 times smaller due to unbounded visible variables. The number of factors F of TvRBM is set to 100. For both RBM and TvRBM, 500 hidden units are used for images and 200 for EEG signals. Hyperparameters are specified using cross-validation. We update parameters after seeing "mini-batches" of $B = 50$ samples. Learning is terminated after 100 scans through the whole data.

We perform classifications to verify the capability of our model. Whenever suitable, the classification errors are compared with the results of standard RBM, matrix methods – 2DPCA (Yang et al. 2004) and 2DNMF (Zhang, Chen, and hua Zhou 2005), and tensor decomposition methods – the Tucker (Tucker 1963) and the PARAFAC (Carroll and

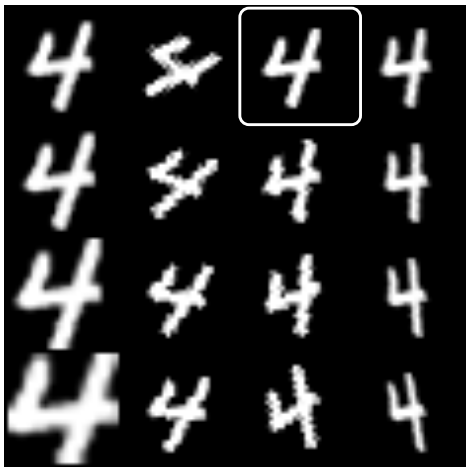

Figure 3: An example of handwritten digit 4: original – bounded one and its 15 distorted versions: zooms in the first column, rotations in the second and third, and horizontal shears in the last one.

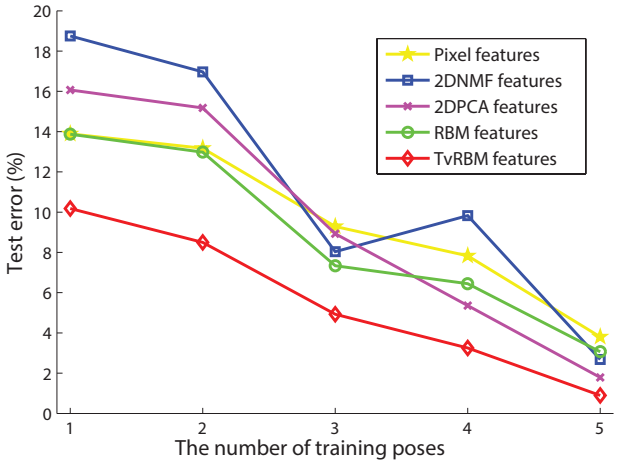


Figure 4: The recognition errors (%) on Extended YaleB dataset. Two modes are 65 illumination conditions and 192 ($12 \times 16$) pixels. Images with 4 facial poses are used for testing.

Chang 1970). We implement the 2DPCA as described in Sec. 2 of (Yang et al. 2004) and the 2DNMF following the Algorithm 2 in Sec. 3.1 of (Zhang, Chen, and hua Zhou 2005). Using these methods, the images are transformed into feature matrix or feature image. For Tucker and PARAFAC, we use implementations in (Andersson and Bro 2000). The EEG data are projected onto lower dimensional tensor and vector representations. The learnt features of matrix and tensor methods are finally concatenated into vectors and then fed to 1-nearest neighbors (1-NN) with cosine similarity measures for classification. For fair comparisons, the lengths of feature vectors are matched with the hidden posteriors of TvRBM and RBM (i.e., 500 for images, 200 for EEG signals).

### 5.2 Handwritten digit classification with augmentation

We use the MNIST dataset which consists of $60,000$ training and $10,000$ test images of digits from 0 to 9 with the size $28 \times 28$. The images are well-aligned and cropped. We create one more dataset by perturbing the original images to obtain more miscellaneous factors of variation. For the first set of experiments (Original), we treat each image as a matrix, hypothesizing that variations in handwriting styles and numbers would moderate the strokes in the horizontal and vertical directions. For the second set of experiments (Augment), we augment each image with 15 distorted versions of varying degree: 4 zooms, 7 rotations and 4 horizontal shears (see Fig. 3 for an illustration). The original image and its variants are then vectorized and stacked to yield a $16 \times 784$ matrix. Ten percent of data with 16-fold augmentation creates a new dataset already 1.6 times larger than the original MNIST. The classification errors on testing data are reported in Tab. 1. It is shown that the TvRBM extracts better discriminative features than its 1D and 2D rivals. The TvRBM also beats RBM on original images (10% error), suggesting a principled way to combine multiple styles of data augmentation, a technique often used to boost performance in vision classification problems.

### 5.3 Face recognition with unseen poses

We use Extended YaleB database (Lee, Ho, and Kriegman 2005) which contains images of 28 subjects under 65 illumination conditions (including the ambient lighting) and 9 poses. We use the original version in which images are neither aligned nor cropped, then subsample images to $12 \times 16$ size. The illumination conditions and pixels are considered as the two modes while poses are separated as data points. We randomly choose 4 facial poses for testing and vary the number of the remaining 5 poses for training. Thus the task is carried under face recognition with *unseen* poses of test images. Fig. 4 shows recognition errors with respect to different numbers of training poses. The performance of the TvRBM is consistently superior to its rivals.

**Visualizing the learnt receptive fields.** We estimate the receptive fields of the TvRBM using Eq. (4). For reference, we compare these filters with those learnt by RBM on individual images. Fig. 5 plots 64 receptive fields for each

| | Classification error (%) | |
|---|---|---|
| Method | Original | Augment(*) |
| Pixel | 2.77 | 5.2 |
| RBM | 2.71 | 4.9 |
| 2DNMF | 2.64 | 5.1 |
| 2DPCA | 2.59 | 4.9 |
| TvRBM | **2.42** | **4.7** |

Table 1: The classification errors (%) on testing parts of original and augmented MNIST images. (*) 10% data is used.

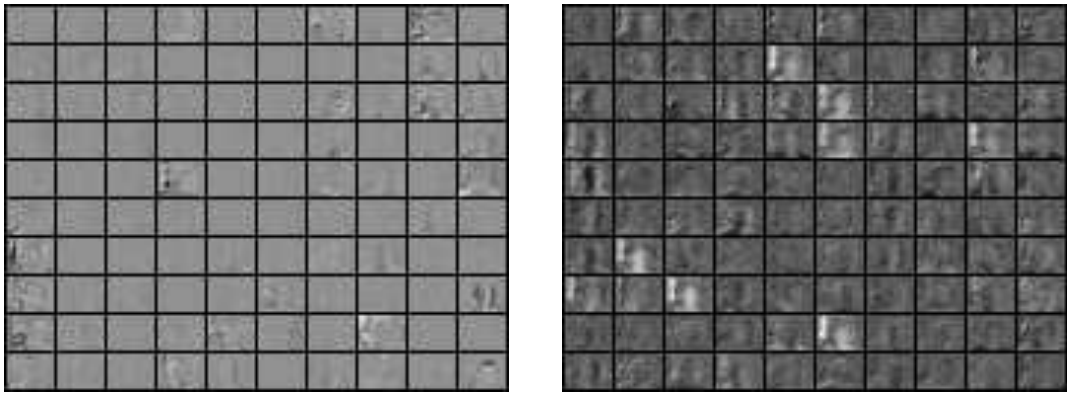

Figure 5: The learnt receptive fields of RBM (left) and TvRBM (right) for facial images with different lighting conditions. The illumination variations are captured in TvRBM filters.

method. The TvRBM produces clear facial filters responding to lighting variations. The RBM, on the other hand, fails to capture the illumination changes and has many "dead" hidden units with no facial information. Note that the unused hidden units problem of RBM is not uncommon which has been studied in (Berglund, Raiko, and Cho 2013).

### 5.4 EEG-based alcoholic diagnosis with unseen subjects

The EEG dataset collected in (Zhang et al. 1995) contains readings of 64 electrodes placed on the scalp of 122 subjects in two (alcoholic and control) groups. For each trial, a subject was presented with visual stimuli and their brain activities were recorded. The signals (in $\mu V$) are sampled at 256Hz for 1 second. These time-series signals are converted into $64 \times 64$ spectrograms using short-time Fourier transform with Hamming window of length 64, 54 overlapping samples. This results in 3-mode tensors of size $64 \times 64 \times 64$. The pixels are normalized across the dataset to obtain zero-means and unit variances.

The objective is to diagnose a subject using a single visual stimulus. Here we use only one trial per subject and randomly select 36 subjects for testing. We compare the results of our model with classic tensor decomposition methods – the Tucker and the PARAFAC. Here all data points are stacked into a 4-mode tensor. The classification performances are shown in Tab. 2. The RBM fails to learn the enormous number of parameters – $200 \times 64^3$. Regardless to the training sizes, the TvRBM achieves better results than the tensor decomposition methods.

| | Classification error (%) | | | | |
|---|---|---|---|---|---|
| Method | 5% | 10% | 25% | 50% | 100% |
| Pixel | 52.78 | 41.67 | 38.89 | 37.24 | 36.11 |
| Tucker | 52.78 | 44.44 | 44.44 | 38.89 | 33.33 |
| PARAFAC | 58.33 | 52.78 | 52.78 | 48.67 | 44.44 |
| RBM | – | – | – | – | – |
| TvRBM | **47.22** | **36.11** | **27.78** | **25.00** | **19.44** |

Table 2: The control/alcoholic classification performance on testing set of EEG data (Zhang et al. 1995) with different portion of training data. The RBM fails to learn from the excessive number of pixels per trial reading.

## 6 Conclusion

We have introduced a novel model called Tensor-variate Restricted Boltzmann Machine (TvRBM) to model distribution of N-mode data. The model inherits many attractive qualities of the RBM family, namely distributed representation, fast extract posterior inference and efficient MCMC-based learning. However, unlike the flat RBM, the TvRBM preserves the multimode structures of the data. Through a (N+1)-way factoring scheme, the multiplicative interactions between modes are moderated by a hidden layer, which, in turn, captures the factors of variation in the data. The model is highly compact: the number of mapping parameters grows linearly rather than exponentially with number of modes, and thus allows learning with limited data. Comprehensive experiments on three real-world applications – handwritten digit classification, face recognition, and EEG-based alcoholic diagnosis – demonstrate that with less parameters, TvRBM is feasible and easier to be learnt, even with very few training samples. The multiplicative instead of additive interactions among data-modes help hidden units of TvRBM capture separate modes (e.g., poses and lighting conditions, progression in time, localized frequency) better, resulting in more robust latent representations than standard RBM and existing multiway models.

The tensor treatment opens up new perspectives for modeling multichannel, multitype data, especially those in healthcare. Our on going work (omitted 2014), for example, models electronic medical records, which involve temporal information of hospital encounters, pathological readings, diagnoses and interventions. The data is multitype in that it includes a mixture of binary, continuous, counts and multinomial components. This poses new kinds of challenge and opportunities for TvRBM.

## References


Andersson, C. A., and Bro, R. 2000. The n-way toolbox for matlab. *Chemometrics and Intelligent Laboratory Systems* 52(1):1–4.

Berglund, M.; Raiko, T.; and Cho, K. 2013. Measuring the usefulness of hidden units in boltzmann machines with mutual information. In *Neural Information Processing (ICONIP)*, volume 8226 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 482–489.

Carroll, J. D., and Chang, J.-J. 1970. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika* 35(3):283–319.

Chu, W., and Ghahramani, Z. 2009. Probabilistic models for incomplete multi-dimensional arrays. In *In Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Hinton, G., and Salakhutdinov, R. 2006. Reducing the dimensionality of data with neural networks. *Science* 313(5786):504 – 507.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural Computation* 14(8):1771–1800.

Kim, Y.-D., and Choi, S. 2007. Nonnegative tucker decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–8. IEEE.

Kolda, T. G., and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM review* 51(3):455–500.

Lee, K.; Ho, J.; and Kriegman, D. 2005. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)* 27(5):684–698.

Memisevic, R., and Hinton, G. 2007. Unsupervised learning of image transformations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–8. IEEE.

Memisevic, R., and Hinton, G. E. 2010. Learning to represent spatial transformations with factored higher-order boltzmann machines. *Neural Computation* 22(6):1473–1492.

omitted, A. 2014. Modeling electronic medical records using multichannel maxtrix-variate restricted Boltzmann machines. *In submission*.

Ranzato, M.; Krizhevsky, A.; and Hinton, G. E. 2010. Factored 3-way restricted boltzmann machines for modeling natural images. In *International Conference on Artificial Intelligence and Statistics (ICAIS)*, 621–628.

Reed, S., and Lee, H. 2013. Learning deep representations via multiplicative interactions between factors of variation. *Advances in Neural Information Processing Systems (NIPS)*.

Salakhutdinov, R., and Hinton, G. 2009a. Deep Boltzmann Machines. In *Proceedings of 20th AISTATS*, volume 5, 448–455.

Salakhutdinov, R., and Hinton, G. 2009b. Semantic hashing. *International Journal of Approximate Reasoning* 50(7):969–978.

Salakhutdinov, R., and Hinton, G. 2009c. Replicated softmax: an undirected topic model. *Advances in Neural Information Processing Systems (NIPS)* 22:1607–1614.

Shashua, A., and Hazan, T. 2005. Non-negative tensor factorization with applications to statistics and computer vision. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, 792–799. ACM.

Smolensky, P. 1986. Parallel distributed processing: explorations in the microstructure of cognition, vol. 1. Cambridge, MA, USA: MIT Press. chapter Information processing in dynamical systems: foundations of harmony theory, 194–281.

Tang, Y.; Salakhutdinov, R.; and Hinton, G. 2013. Tensor analyzers. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*.

Taylor, G. W., and Hinton, G. E. 2009. Factored conditional restricted boltzmann machines for modeling motion style. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, 1025–1032. New York, NY, USA: ACM.

Tieleman, T., and Hinton, G. 2009. Using fast weights to improve persistent contrastive divergence. In *ICML*. ACM New York, NY, USA.

Tran, T.; Phung, D.; and Venkatesh, S. 2013. Thurstonian Boltzmann Machines: Learning from Multiple Inequalities. In *International Conference on Machine Learning (ICML)*.

Tucker, L. R. 1963. Implications of factor analysis of three-way matrices for measurement of change. *Problems in measuring change* 122–137.

Xie, X.; Yan, S.; Kwok, J. T.; and Huang, T. S. 2008. Matrix-variate factor analysis and its applications. *IEEE Transactions on Neural Networks* 19(10):1821–1826.

Xu, Z.; Yan, F.; and Qi, A. 2012. Infinite tucker decomposition: Nonparametric bayesian models for multiway data analysis. In Langford, J., and Pineau, J., eds., *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 1023–1030. New York, NY, USA: ACM.

Yang, J.; Zhang, D.; Frangi, A.; and Yang, J.-Y. 2004. Two-dimensional pca: a new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 26(1):131–137.

Ye, J.; Janardan, R.; and Li, Q. 2004. Two-dimensional linear discriminant analysis. In *Advances in Neural Information Processing Systems (NIPS)*, 1569–1576.

Zhang, X. L.; Begleiter, H.; Porjesz, B.; Wang, W.; and Litke, A. 1995. Event related potentials during object recognition tasks. *Brain Research Bulletin* 38(6):531–538.

Zhang, D.; Chen, S.; and hua Zhou, Z. 2005. Two-dimensional non-negative matrix factorization for face representation and recognition. In *ICCV'05 Workshop on Analysis and Modeling of Faces and Gestures (AMFG)*, 350–363. Springer.